

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims

1. (Currently Amended) A method for allocating resources, comprising:
before receiving a request from a super process that needs resources from a first cluster and a second cluster,
at ~~[[a]] the first cluster and a second cluster,~~ allocating reserved resources to two or more depth levels, wherein the reserved resources form one or more reserved pools at each of the two or more depth levels; and
at the second cluster, allocating reserved resources to two or more depth levels, wherein the reserved resources form one or more reserved pools at each of the two or more depth levels; and
at the first cluster, upon receiving a request for allocation of resources from ~~[[a]] the super process,~~ determining a depth level from which to allocate resources by:
determining whether the request is a remote request from the second cluster or a local request from the first cluster; and
in response to determining that the request is the remote request and that resources are being allocated from a ~~[[first]]~~ current depth level to the process at the second cluster,
determining that the depth level at the first cluster is a next depth level that is an increment from the ~~[[first]]~~ current depth level at the second cluster, wherein each time the process issues ~~[[a]] an inter cluster request from one of the first cluster and the second cluster to another of the first cluster and the second cluster,~~ the depth level is incremented by one until reaching a maximum depth level to avoid a deadlock situation; ~~[[and]]~~
attempting to allocate a reserved pool from the determined depth level at the first cluster;

in response to the attempted allocation of the reserved pool being unsuccessful for the remote request, attempting to allocate resources from an unreserved pool; and

in response to the attempted allocation from the unreserved pool being unsuccessful, placing the request in a data structure to wait for the reserved pool.

2. (Original) The method of claim 1, further comprising:
generating control structures that indicate which resources are allocated to which processes.
3. (Previously Presented) The method of claim 1, wherein the allocations occur at the first cluster and further comprising:
at the first cluster, waiting for the second cluster to finish initialization processing that allocates reserved resources of the second cluster to each of multiple depth levels before allowing requests for resources to be processed at the first cluster.
4. (Cancelled) The method of claim 1, further comprising:
when the attempted allocation of the reserved pool is unsuccessful for the remote request, attempting to allocate resources from an unreserved pool.
5. (Cancelled) The method of claim 4, further comprising:
when the attempted allocation from the unreserved pool is unsuccessful, placing the request in a data structure to wait for a reserved pool.
6. (Original) The method of claim 1, wherein the resources are task control blocks.
7. (Previously Presented) The method of claim 1, further comprising:
determining that the request is the local request and that a reserved pool at a particular depth level has been allocated to the process at the first cluster; and
allocating a resource from the reserved pool at the first cluster.

8-9. (Cancelled)

10. (Previously Presented) The method of claim 1, further comprising:
determining that processing with the resource is complete; and
returning the resource to a pool of resources.

11. (Original) The method of claim 10, further comprising:
when the resource is returned to a reserved pool, determining whether all resources have
been returned to that reserved pool;
when all resources have been returned, freeing the reserved pool for allocation to another
process; and
allocating the freed reserved pool to a request waiting for allocation of a reserved pool.

12. (Original) The method of claim 10, further comprising:
when the resource is returned to an unreserved pool, allocating the freed unreserved pool
to a request waiting for allocation of a reserved pool at a current depth level.

13-30. (Cancelled)

31. (Currently Amended) An article of manufacture comprising a computer readable
medium storing code for allocating resources, wherein the code when executed by a processor of
a computer causes operations to be performed, the operations comprising:

before receiving a request from a super process that needs resources from a first cluster
and a second cluster,

at [[a]] the first cluster and a second cluster, allocating reserved resources to two
or more depth levels, wherein the reserved resources form one or more reserved pools at
each of the two or more depth levels;

at the second cluster, allocating reserved resources to two or more depth levels,
wherein the reserved resources form one or more reserved pools at each of the two or
more depth levels; and

at the first cluster, upon receiving a request for allocation of resources from [[a]] the super process, determining a depth level from which to allocate resources by:

determining whether the request is a remote request from the second cluster or a local request from the first cluster; and

in response to determining that the request is the remote request and that resources are being allocated from a [[first]] current depth level to the process at the second cluster,

determining that the depth level at the first cluster is a next depth level that is an increment from the [[first]] current depth level at the second cluster, wherein each time the process issues [[a]] an inter cluster request ~~from one of the first cluster and the second cluster to another of the first cluster and the second cluster~~, the depth level is incremented by one until reaching a maximum depth level to avoid a deadlock situation; [[and]]

attempting to allocate a reserved pool from the determined depth level at the first cluster;

in response to the attempted allocation of the reserved pool being unsuccessful for the remote request, attempting to allocate resources from an unreserved pool; and

in response to the attempted allocation from the unreserved pool being unsuccessful, placing the request in a data structure to wait for the reserved pool.

32. (New) The article of manufacture of claim 31, wherein the operations further comprise:

generating control structures that indicate which resources are allocated to which processes.

33. (New) The article of manufacture of claim 31, wherein the allocations occur at the first cluster and wherein the operations further comprise:

at the first cluster, waiting for the second cluster to finish initialization processing that allocates reserved resources of the second cluster to each of multiple depth levels before allowing requests for resources to be processed at the first cluster.

34. (Cancelled)

35. (Cancelled)

36. (New) The article of manufacture of claim 31, wherein the resources are task control blocks.

37. (New) The article of manufacture of claim 31, wherein the operations further comprise:

determining that the request is the local request and that a reserved pool at a particular depth level has been allocated to the process at the first cluster; and
allocating a resource from the reserved pool at the first cluster.

38. (New) The article of manufacture of claim 31, wherein the operations further comprise:

determining that processing with the resource is complete; and
returning the resource to a pool of resources.

39. (New) The article of manufacture of claim 38, wherein the operations further comprise:

when the resource is returned to a reserved pool, determining whether all resources have been returned to that reserved pool;

when all resources have been returned, freeing the reserved pool for allocation to another process; and

allocating the freed reserved pool to a request waiting for allocation of a reserved pool.

40. (New) The article of manufacture of claim 38, wherein the operations further comprise:

when the resource is returned to an unreserved pool, allocating the freed unreserved pool to a request waiting for allocation of a reserved pool at a current depth level.

41. (Currently Amended) A system including circuitry for allocating resources, wherein the circuitry is capable of causing operations to be performed, the operations comprising:

before receiving a request from a super process that needs resources from a first cluster and a second cluster,

at ~~[[a]] the first cluster and a second cluster,~~ allocating reserved resources to two or more depth levels, wherein the reserved resources form one or more reserved pools at each of the two or more depth levels; and

at the second cluster, allocating reserved resources to two or more depth levels, wherein the reserved resources form one or more reserved pools at each of the two or more depth levels; and

at the first cluster, upon receiving a request for allocation of resources from ~~[[a]] the super process,~~ determining a depth level from which to allocate resources by:

determining whether the request is a remote request from the second cluster or a local request from the first cluster; and

in response to determining that the request is the remote request and that resources are being allocated from a ~~[[first]]~~ current depth level to the process at the second cluster,

determining that the depth level at the first cluster is a next depth level that is an increment from the ~~[[first]]~~ current depth level at the second cluster, wherein each time the process issues ~~[[a]] an inter cluster request from one of the first cluster and the second cluster to another of the first cluster and the second cluster,~~ the depth level is incremented by one until reaching a maximum depth level to avoid a deadlock situation; ~~[[and]]~~

attempting to allocate a reserved pool from the determined depth level at the first cluster;

in response to the attempted allocation of the reserved pool being unsuccessful for the remote request, attempting to allocate resources from an unreserved pool; and

in response to the attempted allocation from the unreserved pool being unsuccessful, placing the request in a data structure to wait for the reserved pool.

42. (New) The system of claim 41, wherein the operations further comprise: generating control structures that indicate which resources are allocated to which processes.

43. (New) The system of claim 41, wherein the operations for the allocations occur at the first cluster and further comprise:

at the first cluster, waiting for the second cluster to finish initialization processing that allocates reserved resources of the second cluster to each of multiple depth levels before allowing requests for resources to be processed at the first cluster.

44. (Cancelled)

45. (Cancelled)

46. (New) The system of claim 44, wherein the resources are task control blocks.

47. (New) The system of claim 41, wherein the operations further comprise: determining that the request is the local request and that a reserved pool at a particular depth level has been allocated to the process at the first cluster; and allocating a resource from the reserved pool at the first cluster.

48. (New) The system of claim 41, wherein the operations further comprise: determining that processing with the resource is complete; and returning the resource to a pool of resources.

49. (New) The system of claim 48, wherein the operations further comprise:

when the resource is returned to a reserved pool, determining whether all resources have been returned to that reserved pool;

when all resources have been returned, freeing the reserved pool for allocation to another process; and

allocating the freed reserved pool to a request waiting for allocation of a reserved pool.

50. (New) The system of claim 48, wherein the operations further comprise:

when the resource is returned to an unreserved pool, allocating the freed unreserved pool to a request waiting for allocation of a reserved pool at a current depth level.